| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/BasicSplitPaneUI.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneDivider.VerticalDragController.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.BasicHorizontalLayoutManager.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/plaf/basic/BasicSplitPaneUI.html)    [**NO FRAMES**](http://docs.google.com/BasicSplitPaneUI.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#3znysh7) | [FIELD](#2et92p0) | [CONSTR](#tyjcwt) | [METHOD](#3dy6vkm) | DETAIL: [FIELD](#2s8eyo1) | [CONSTR](#23ckvvd) | [METHOD](#32hioqz) |

## **javax.swing.plaf.basic**

Class BasicSplitPaneUI

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 [javax.swing.plaf.ComponentUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html)  
 [javax.swing.plaf.SplitPaneUI](http://docs.google.com/javax/swing/plaf/SplitPaneUI.html)  
 **javax.swing.plaf.basic.BasicSplitPaneUI**

**Direct Known Subclasses:** [MetalSplitPaneUI](http://docs.google.com/javax/swing/plaf/metal/MetalSplitPaneUI.html)

public class **BasicSplitPaneUI**extends [SplitPaneUI](http://docs.google.com/javax/swing/plaf/SplitPaneUI.html)

A Basic L&F implementation of the SplitPaneUI.

| **Nested Class Summary** | |
| --- | --- |
| class | [**BasicSplitPaneUI.BasicHorizontalLayoutManager**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.BasicHorizontalLayoutManager.html)            LayoutManager for JSplitPanes that have an orientation of HORIZONTAL\_SPLIT. |
| class | [**BasicSplitPaneUI.BasicVerticalLayoutManager**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.BasicVerticalLayoutManager.html)            LayoutManager used for JSplitPanes with an orientation of VERTICAL\_SPLIT. |
| class | [**BasicSplitPaneUI.FocusHandler**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.FocusHandler.html)            Implementation of the FocusListener that the JSplitPane UI uses. |
| class | [**BasicSplitPaneUI.KeyboardDownRightHandler**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.KeyboardDownRightHandler.html)            Implementation of an ActionListener that the JSplitPane UI uses for handling specific key presses. |
| class | [**BasicSplitPaneUI.KeyboardEndHandler**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.KeyboardEndHandler.html)            Implementation of an ActionListener that the JSplitPane UI uses for handling specific key presses. |
| class | [**BasicSplitPaneUI.KeyboardHomeHandler**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.KeyboardHomeHandler.html)            Implementation of an ActionListener that the JSplitPane UI uses for handling specific key presses. |
| class | [**BasicSplitPaneUI.KeyboardResizeToggleHandler**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.KeyboardResizeToggleHandler.html)            Implementation of an ActionListener that the JSplitPane UI uses for handling specific key presses. |
| class | [**BasicSplitPaneUI.KeyboardUpLeftHandler**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.KeyboardUpLeftHandler.html)            Implementation of an ActionListener that the JSplitPane UI uses for handling specific key presses. |
| class | [**BasicSplitPaneUI.PropertyHandler**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.PropertyHandler.html)            Implementation of the PropertyChangeListener that the JSplitPane UI uses. |

| **Field Summary** | |
| --- | --- |
| protected  int | [**beginDragDividerLocation**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#beginDragDividerLocation)            Location of the divider when the dragging session began. |
| protected  [BasicSplitPaneDivider](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneDivider.html) | [**divider**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#divider)            Instance of the divider for this JSplitPane. |
| protected  [KeyStroke](http://docs.google.com/javax/swing/KeyStroke.html) | [**dividerResizeToggleKey**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#dividerResizeToggleKey)  **Deprecated.** *As of Java 2 platform v1.3.* |
| protected  int | [**dividerSize**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#dividerSize)            The size of the divider while the dragging session is valid. |
| protected  [KeyStroke](http://docs.google.com/javax/swing/KeyStroke.html) | [**downKey**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#downKey)  **Deprecated.** *As of Java 2 platform v1.3.* |
| protected  boolean | [**draggingHW**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#draggingHW)            Set to true in startDragging if any of the children (not including the nonContinuousLayoutDivider) are heavy weights. |
| protected  [KeyStroke](http://docs.google.com/javax/swing/KeyStroke.html) | [**endKey**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#endKey)  **Deprecated.** *As of Java 2 platform v1.3.* |
| protected  [FocusListener](http://docs.google.com/java/awt/event/FocusListener.html) | [**focusListener**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#focusListener)            Instance of the FocusListener for this JSplitPane. |
| protected  [KeyStroke](http://docs.google.com/javax/swing/KeyStroke.html) | [**homeKey**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#homeKey)  **Deprecated.** *As of Java 2 platform v1.3.* |
| protected static int | [**KEYBOARD\_DIVIDER\_MOVE\_OFFSET**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#KEYBOARD_DIVIDER_MOVE_OFFSET)            How far (relative) the divider does move when it is moved around by the cursor keys on the keyboard. |
| protected  [ActionListener](http://docs.google.com/java/awt/event/ActionListener.html) | [**keyboardDownRightListener**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#keyboardDownRightListener)  **Deprecated.** *As of Java 2 platform v1.3.* |
| protected  [ActionListener](http://docs.google.com/java/awt/event/ActionListener.html) | [**keyboardEndListener**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#keyboardEndListener)  **Deprecated.** *As of Java 2 platform v1.3.* |
| protected  [ActionListener](http://docs.google.com/java/awt/event/ActionListener.html) | [**keyboardHomeListener**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#keyboardHomeListener)  **Deprecated.** *As of Java 2 platform v1.3.* |
| protected  [ActionListener](http://docs.google.com/java/awt/event/ActionListener.html) | [**keyboardResizeToggleListener**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#keyboardResizeToggleListener)  **Deprecated.** *As of Java 2 platform v1.3.* |
| protected  [ActionListener](http://docs.google.com/java/awt/event/ActionListener.html) | [**keyboardUpLeftListener**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#keyboardUpLeftListener)  **Deprecated.** *As of Java 2 platform v1.3.* |
| protected  [BasicSplitPaneUI.BasicHorizontalLayoutManager](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.BasicHorizontalLayoutManager.html) | [**layoutManager**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#layoutManager)            LayoutManager that is created and placed into the split pane. |
| protected  [KeyStroke](http://docs.google.com/javax/swing/KeyStroke.html) | [**leftKey**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#leftKey)  **Deprecated.** *As of Java 2 platform v1.3.* |
| protected static [String](http://docs.google.com/java/lang/String.html) | [**NON\_CONTINUOUS\_DIVIDER**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#NON_CONTINUOUS_DIVIDER)            The divider used for non-continuous layout is added to the split pane with this object. |
| protected  [Component](http://docs.google.com/java/awt/Component.html) | [**nonContinuousLayoutDivider**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#nonContinuousLayoutDivider)            Instance for the shadow of the divider when non continuous layout is being used. |
| protected  [PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html) | [**propertyChangeListener**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#propertyChangeListener)            Instance of the PropertyChangeListener for this JSplitPane. |
| protected  [KeyStroke](http://docs.google.com/javax/swing/KeyStroke.html) | [**rightKey**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#rightKey)  **Deprecated.** *As of Java 2 platform v1.3.* |
| protected  [JSplitPane](http://docs.google.com/javax/swing/JSplitPane.html) | [**splitPane**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#splitPane)            JSplitPane instance this instance is providing the look and feel for. |
| protected  [KeyStroke](http://docs.google.com/javax/swing/KeyStroke.html) | [**upKey**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#upKey)  **Deprecated.** *As of Java 2 platform v1.3.* |

| **Constructor Summary** | |
| --- | --- |
| [**BasicSplitPaneUI**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#BasicSplitPaneUI())() |

| **Method Summary** | |
| --- | --- |
| [BasicSplitPaneDivider](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneDivider.html) | [**createDefaultDivider**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#createDefaultDivider())()            Creates the default divider. |
| protected  [Component](http://docs.google.com/java/awt/Component.html) | [**createDefaultNonContinuousLayoutDivider**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#createDefaultNonContinuousLayoutDivider())()            Returns the default non continuous layout divider, which is an instanceof Canvas that fills the background in dark gray. |
| protected  [FocusListener](http://docs.google.com/java/awt/event/FocusListener.html) | [**createFocusListener**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#createFocusListener())()            Creates a FocusListener for the JSplitPane UI. |
| protected  [ActionListener](http://docs.google.com/java/awt/event/ActionListener.html) | [**createKeyboardDownRightListener**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#createKeyboardDownRightListener())()  **Deprecated.** *As of Java 2 platform v1.3.* |
| protected  [ActionListener](http://docs.google.com/java/awt/event/ActionListener.html) | [**createKeyboardEndListener**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#createKeyboardEndListener())()  **Deprecated.** *As of Java 2 platform v1.3.* |
| protected  [ActionListener](http://docs.google.com/java/awt/event/ActionListener.html) | [**createKeyboardHomeListener**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#createKeyboardHomeListener())()  **Deprecated.** *As of Java 2 platform v1.3.* |
| protected  [ActionListener](http://docs.google.com/java/awt/event/ActionListener.html) | [**createKeyboardResizeToggleListener**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#createKeyboardResizeToggleListener())()  **Deprecated.** *As of Java 2 platform v1.3.* |
| protected  [ActionListener](http://docs.google.com/java/awt/event/ActionListener.html) | [**createKeyboardUpLeftListener**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#createKeyboardUpLeftListener())()  **Deprecated.** *As of Java 2 platform v1.3.* |
| protected  [PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html) | [**createPropertyChangeListener**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#createPropertyChangeListener())()            Creates a PropertyChangeListener for the JSplitPane UI. |
| static [ComponentUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html) | [**createUI**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#createUI(javax.swing.JComponent))([JComponent](http://docs.google.com/javax/swing/JComponent.html) x)            Creates a new BasicSplitPaneUI instance |
| protected  void | [**dragDividerTo**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#dragDividerTo(int))(int location)            Messaged during a dragging session to move the divider to the passed in location. |
| protected  void | [**finishDraggingTo**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#finishDraggingTo(int))(int location)            Messaged to finish the dragging session. |
| void | [**finishedPaintingChildren**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#finishedPaintingChildren(javax.swing.JSplitPane,%20java.awt.Graphics))([JSplitPane](http://docs.google.com/javax/swing/JSplitPane.html) jc, [Graphics](http://docs.google.com/java/awt/Graphics.html) g)            Messaged after the JSplitPane the receiver is providing the look and feel for paints its children. |
| [BasicSplitPaneDivider](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneDivider.html) | [**getDivider**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#getDivider())()            Returns the divider between the top Components. |
| protected  int | [**getDividerBorderSize**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#getDividerBorderSize())()  **Deprecated.** *As of Java 2 platform v1.3, instead set the border on the divider.* |
| int | [**getDividerLocation**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#getDividerLocation(javax.swing.JSplitPane))([JSplitPane](http://docs.google.com/javax/swing/JSplitPane.html) jc)            Returns the location of the divider, which may differ from what the splitpane thinks the location of the divider is. |
| [Insets](http://docs.google.com/java/awt/Insets.html) | [**getInsets**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#getInsets(javax.swing.JComponent))([JComponent](http://docs.google.com/javax/swing/JComponent.html) jc)            Returns the insets. |
| int | [**getLastDragLocation**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#getLastDragLocation())()            Returns the last drag location of the JSplitPane. |
| int | [**getMaximumDividerLocation**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#getMaximumDividerLocation(javax.swing.JSplitPane))([JSplitPane](http://docs.google.com/javax/swing/JSplitPane.html) jc)            Gets the maximum location of the divider. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | [**getMaximumSize**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#getMaximumSize(javax.swing.JComponent))([JComponent](http://docs.google.com/javax/swing/JComponent.html) jc)            Returns the maximum size for the passed in component, This is passed off to the current layoutmanager. |
| int | [**getMinimumDividerLocation**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#getMinimumDividerLocation(javax.swing.JSplitPane))([JSplitPane](http://docs.google.com/javax/swing/JSplitPane.html) jc)            Gets the minimum location of the divider. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | [**getMinimumSize**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#getMinimumSize(javax.swing.JComponent))([JComponent](http://docs.google.com/javax/swing/JComponent.html) jc)            Returns the minimum size for the passed in component, This is passed off to the current layoutmanager. |
| [Component](http://docs.google.com/java/awt/Component.html) | [**getNonContinuousLayoutDivider**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#getNonContinuousLayoutDivider())()            Returns the divider to use when the splitPane is configured to not continuously layout. |
| int | [**getOrientation**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#getOrientation())()            Returns the orientation for the JSplitPane. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | [**getPreferredSize**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#getPreferredSize(javax.swing.JComponent))([JComponent](http://docs.google.com/javax/swing/JComponent.html) jc)            Returns the preferred size for the passed in component, This is passed off to the current layoutmanager. |
| [JSplitPane](http://docs.google.com/javax/swing/JSplitPane.html) | [**getSplitPane**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#getSplitPane())()            Returns the splitpane this instance is currently contained in. |
| protected  void | [**installDefaults**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#installDefaults())()            Installs the UI defaults. |
| protected  void | [**installKeyboardActions**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#installKeyboardActions())()            Installs the keyboard actions for the UI. |
| protected  void | [**installListeners**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#installListeners())()            Installs the event listeners for the UI. |
| void | [**installUI**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#installUI(javax.swing.JComponent))([JComponent](http://docs.google.com/javax/swing/JComponent.html) c)            Installs the UI. |
| boolean | [**isContinuousLayout**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#isContinuousLayout())()            Determines wether the JSplitPane is set to use a continuous layout. |
| void | [**paint**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#paint(java.awt.Graphics,%20javax.swing.JComponent))([Graphics](http://docs.google.com/java/awt/Graphics.html) g, [JComponent](http://docs.google.com/javax/swing/JComponent.html) jc)            Messaged to paint the look and feel. |
| protected  void | [**resetLayoutManager**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#resetLayoutManager())()            Resets the layout manager based on orientation and messages it with invalidateLayout to pull in appropriate Components. |
| void | [**resetToPreferredSizes**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#resetToPreferredSizes(javax.swing.JSplitPane))([JSplitPane](http://docs.google.com/javax/swing/JSplitPane.html) jc)            Messaged to reset the preferred sizes. |
| void | [**setContinuousLayout**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#setContinuousLayout(boolean))(boolean b)            Turn continuous layout on/off. |
| void | [**setDividerLocation**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#setDividerLocation(javax.swing.JSplitPane,%20int))([JSplitPane](http://docs.google.com/javax/swing/JSplitPane.html) jc, int location)            Sets the location of the divider to location. |
| void | [**setLastDragLocation**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#setLastDragLocation(int))(int l)            Set the last drag location of the JSplitPane. |
| protected  void | [**setNonContinuousLayoutDivider**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#setNonContinuousLayoutDivider(java.awt.Component))([Component](http://docs.google.com/java/awt/Component.html) newDivider)            Sets the divider to use when the splitPane is configured to not continuously layout. |
| protected  void | [**setNonContinuousLayoutDivider**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#setNonContinuousLayoutDivider(java.awt.Component,%20boolean))([Component](http://docs.google.com/java/awt/Component.html) newDivider, boolean rememberSizes)            Sets the divider to use. |
| void | [**setOrientation**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#setOrientation(int))(int orientation)            Set the orientation for the JSplitPane. |
| protected  void | [**startDragging**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#startDragging())()            Should be messaged before the dragging session starts, resets lastDragLocation and dividerSize. |
| protected  void | [**uninstallDefaults**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#uninstallDefaults())()            Uninstalls the UI defaults. |
| protected  void | [**uninstallKeyboardActions**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#uninstallKeyboardActions())()            Uninstalls the keyboard actions for the UI. |
| protected  void | [**uninstallListeners**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#uninstallListeners())()            Uninstalls the event listeners for the UI. |
| void | [**uninstallUI**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.html#uninstallUI(javax.swing.JComponent))([JComponent](http://docs.google.com/javax/swing/JComponent.html) c)            Uninstalls the UI. |

| **Methods inherited from class javax.swing.plaf.**[**ComponentUI**](http://docs.google.com/javax/swing/plaf/ComponentUI.html) |
| --- |
| [contains](http://docs.google.com/javax/swing/plaf/ComponentUI.html#contains(javax.swing.JComponent,%20int,%20int)), [getAccessibleChild](http://docs.google.com/javax/swing/plaf/ComponentUI.html#getAccessibleChild(javax.swing.JComponent,%20int)), [getAccessibleChildrenCount](http://docs.google.com/javax/swing/plaf/ComponentUI.html#getAccessibleChildrenCount(javax.swing.JComponent)), [getBaseline](http://docs.google.com/javax/swing/plaf/ComponentUI.html#getBaseline(javax.swing.JComponent,%20int,%20int)), [getBaselineResizeBehavior](http://docs.google.com/javax/swing/plaf/ComponentUI.html#getBaselineResizeBehavior(javax.swing.JComponent)), [update](http://docs.google.com/javax/swing/plaf/ComponentUI.html#update(java.awt.Graphics,%20javax.swing.JComponent)) |

| **Methods inherited from class java.lang.**[**Object**](http://docs.google.com/java/lang/Object.html) |
| --- |
| [clone](http://docs.google.com/java/lang/Object.html#clone()), [equals](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object)), [finalize](http://docs.google.com/java/lang/Object.html#finalize()), [getClass](http://docs.google.com/java/lang/Object.html#getClass()), [hashCode](http://docs.google.com/java/lang/Object.html#hashCode()), [notify](http://docs.google.com/java/lang/Object.html#notify()), [notifyAll](http://docs.google.com/java/lang/Object.html#notifyAll()), [toString](http://docs.google.com/java/lang/Object.html#toString()), [wait](http://docs.google.com/java/lang/Object.html#wait()), [wait](http://docs.google.com/java/lang/Object.html#wait(long)), [wait](http://docs.google.com/java/lang/Object.html#wait(long,%20int)) |

| **Field Detail** |
| --- |

### NON\_CONTINUOUS\_DIVIDER

protected static final [String](http://docs.google.com/java/lang/String.html) **NON\_CONTINUOUS\_DIVIDER**

The divider used for non-continuous layout is added to the split pane with this object.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#javax.swing.plaf.basic.BasicSplitPaneUI.NON_CONTINUOUS_DIVIDER)

### KEYBOARD\_DIVIDER\_MOVE\_OFFSET

protected static int **KEYBOARD\_DIVIDER\_MOVE\_OFFSET**

How far (relative) the divider does move when it is moved around by the cursor keys on the keyboard.

### splitPane

protected [JSplitPane](http://docs.google.com/javax/swing/JSplitPane.html) **splitPane**

JSplitPane instance this instance is providing the look and feel for.

### layoutManager

protected [BasicSplitPaneUI.BasicHorizontalLayoutManager](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.BasicHorizontalLayoutManager.html) **layoutManager**

LayoutManager that is created and placed into the split pane.

### divider

protected [BasicSplitPaneDivider](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneDivider.html) **divider**

Instance of the divider for this JSplitPane.

### propertyChangeListener

protected [PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html) **propertyChangeListener**

Instance of the PropertyChangeListener for this JSplitPane.

### focusListener

protected [FocusListener](http://docs.google.com/java/awt/event/FocusListener.html) **focusListener**

Instance of the FocusListener for this JSplitPane.

### dividerSize

protected int **dividerSize**

The size of the divider while the dragging session is valid.

### nonContinuousLayoutDivider

protected [Component](http://docs.google.com/java/awt/Component.html) **nonContinuousLayoutDivider**

Instance for the shadow of the divider when non continuous layout is being used.

### draggingHW

protected boolean **draggingHW**

Set to true in startDragging if any of the children (not including the nonContinuousLayoutDivider) are heavy weights.

### beginDragDividerLocation

protected int **beginDragDividerLocation**

Location of the divider when the dragging session began.

### upKey

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
protected [KeyStroke](http://docs.google.com/javax/swing/KeyStroke.html) **upKey**

**Deprecated.** *As of Java 2 platform v1.3.*As of Java 2 platform v1.3 this previously undocumented field is no longer used. Key bindings are now defined by the LookAndFeel, please refer to the key bindings specification for further details.

### downKey

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
protected [KeyStroke](http://docs.google.com/javax/swing/KeyStroke.html) **downKey**

**Deprecated.** *As of Java 2 platform v1.3.*As of Java 2 platform v1.3 this previously undocumented field is no longer used. Key bindings are now defined by the LookAndFeel, please refer to the key bindings specification for further details.

### leftKey

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
protected [KeyStroke](http://docs.google.com/javax/swing/KeyStroke.html) **leftKey**

**Deprecated.** *As of Java 2 platform v1.3.*As of Java 2 platform v1.3 this previously undocumented field is no longer used. Key bindings are now defined by the LookAndFeel, please refer to the key bindings specification for further details.

### rightKey

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
protected [KeyStroke](http://docs.google.com/javax/swing/KeyStroke.html) **rightKey**

**Deprecated.** *As of Java 2 platform v1.3.*As of Java 2 platform v1.3 this previously undocumented field is no longer used. Key bindings are now defined by the LookAndFeel, please refer to the key bindings specification for further details.

### homeKey

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
protected [KeyStroke](http://docs.google.com/javax/swing/KeyStroke.html) **homeKey**

**Deprecated.** *As of Java 2 platform v1.3.*As of Java 2 platform v1.3 this previously undocumented field is no longer used. Key bindings are now defined by the LookAndFeel, please refer to the key bindings specification for further details.

### endKey

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
protected [KeyStroke](http://docs.google.com/javax/swing/KeyStroke.html) **endKey**

**Deprecated.** *As of Java 2 platform v1.3.*As of Java 2 platform v1.3 this previously undocumented field is no longer used. Key bindings are now defined by the LookAndFeel, please refer to the key bindings specification for further details.

### dividerResizeToggleKey

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
protected [KeyStroke](http://docs.google.com/javax/swing/KeyStroke.html) **dividerResizeToggleKey**

**Deprecated.** *As of Java 2 platform v1.3.*As of Java 2 platform v1.3 this previously undocumented field is no longer used. Key bindings are now defined by the LookAndFeel, please refer to the key bindings specification for further details.

### keyboardUpLeftListener

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
protected [ActionListener](http://docs.google.com/java/awt/event/ActionListener.html) **keyboardUpLeftListener**

**Deprecated.** *As of Java 2 platform v1.3.*As of Java 2 platform v1.3 this previously undocumented field is no longer used. Key bindings are now defined by the LookAndFeel, please refer to the key bindings specification for further details.

### keyboardDownRightListener

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
protected [ActionListener](http://docs.google.com/java/awt/event/ActionListener.html) **keyboardDownRightListener**

**Deprecated.** *As of Java 2 platform v1.3.*As of Java 2 platform v1.3 this previously undocumented field is no longer used. Key bindings are now defined by the LookAndFeel, please refer to the key bindings specification for further details.

### keyboardHomeListener

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
protected [ActionListener](http://docs.google.com/java/awt/event/ActionListener.html) **keyboardHomeListener**

**Deprecated.** *As of Java 2 platform v1.3.*As of Java 2 platform v1.3 this previously undocumented field is no longer used. Key bindings are now defined by the LookAndFeel, please refer to the key bindings specification for further details.

### keyboardEndListener

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
protected [ActionListener](http://docs.google.com/java/awt/event/ActionListener.html) **keyboardEndListener**

**Deprecated.** *As of Java 2 platform v1.3.*As of Java 2 platform v1.3 this previously undocumented field is no longer used. Key bindings are now defined by the LookAndFeel, please refer to the key bindings specification for further details.

### keyboardResizeToggleListener

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
protected [ActionListener](http://docs.google.com/java/awt/event/ActionListener.html) **keyboardResizeToggleListener**

**Deprecated.** *As of Java 2 platform v1.3.*As of Java 2 platform v1.3 this previously undocumented field is no longer used. Key bindings are now defined by the LookAndFeel, please refer to the key bindings specification for further details.

| **Constructor Detail** |
| --- |

### BasicSplitPaneUI

public **BasicSplitPaneUI**()

| **Method Detail** |
| --- |

### createUI

public static [ComponentUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html) **createUI**([JComponent](http://docs.google.com/javax/swing/JComponent.html) x)

Creates a new BasicSplitPaneUI instance

### installUI

public void **installUI**([JComponent](http://docs.google.com/javax/swing/JComponent.html) c)

Installs the UI.

**Overrides:**[installUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html#installUI(javax.swing.JComponent)) in class [ComponentUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html) **Parameters:**c - the component where this UI delegate is being installed**See Also:**[ComponentUI.uninstallUI(javax.swing.JComponent)](http://docs.google.com/javax/swing/plaf/ComponentUI.html#uninstallUI(javax.swing.JComponent)), [JComponent.setUI(javax.swing.plaf.ComponentUI)](http://docs.google.com/javax/swing/JComponent.html#setUI(javax.swing.plaf.ComponentUI)), [JComponent.updateUI()](http://docs.google.com/javax/swing/JComponent.html#updateUI())

### installDefaults

protected void **installDefaults**()

Installs the UI defaults.

### installListeners

protected void **installListeners**()

Installs the event listeners for the UI.

### installKeyboardActions

protected void **installKeyboardActions**()

Installs the keyboard actions for the UI.

### uninstallUI

public void **uninstallUI**([JComponent](http://docs.google.com/javax/swing/JComponent.html) c)

Uninstalls the UI.

**Overrides:**[uninstallUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html#uninstallUI(javax.swing.JComponent)) in class [ComponentUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html) **Parameters:**c - the component from which this UI delegate is being removed; this argument is often ignored, but might be used if the UI object is stateless and shared by multiple components**See Also:**[ComponentUI.installUI(javax.swing.JComponent)](http://docs.google.com/javax/swing/plaf/ComponentUI.html#installUI(javax.swing.JComponent)), [JComponent.updateUI()](http://docs.google.com/javax/swing/JComponent.html#updateUI())

### uninstallDefaults

protected void **uninstallDefaults**()

Uninstalls the UI defaults.

### uninstallListeners

protected void **uninstallListeners**()

Uninstalls the event listeners for the UI.

### uninstallKeyboardActions

protected void **uninstallKeyboardActions**()

Uninstalls the keyboard actions for the UI.

### createPropertyChangeListener

protected [PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html) **createPropertyChangeListener**()

Creates a PropertyChangeListener for the JSplitPane UI.

### createFocusListener

protected [FocusListener](http://docs.google.com/java/awt/event/FocusListener.html) **createFocusListener**()

Creates a FocusListener for the JSplitPane UI.

### createKeyboardUpLeftListener

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
protected [ActionListener](http://docs.google.com/java/awt/event/ActionListener.html) **createKeyboardUpLeftListener**()

**Deprecated.** *As of Java 2 platform v1.3.*

As of Java 2 platform v1.3 this method is no longer used. Subclassers previously using this method should instead create an Action wrapping the ActionListener, and register that Action by overriding installKeyboardActions and placing the Action in the SplitPane's ActionMap. Please refer to the key bindings specification for further details.

Creates a ActionListener for the JSplitPane UI that listens for specific key presses.

### createKeyboardDownRightListener

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
protected [ActionListener](http://docs.google.com/java/awt/event/ActionListener.html) **createKeyboardDownRightListener**()

**Deprecated.** *As of Java 2 platform v1.3.*

As of Java 2 platform v1.3 this method is no longer used. Subclassers previously using this method should instead create an Action wrapping the ActionListener, and register that Action by overriding installKeyboardActions and placing the Action in the SplitPane's ActionMap. Please refer to the key bindings specification for further details.

Creates a ActionListener for the JSplitPane UI that listens for specific key presses.

### createKeyboardHomeListener

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
protected [ActionListener](http://docs.google.com/java/awt/event/ActionListener.html) **createKeyboardHomeListener**()

**Deprecated.** *As of Java 2 platform v1.3.*

As of Java 2 platform v1.3 this method is no longer used. Subclassers previously using this method should instead create an Action wrapping the ActionListener, and register that Action by overriding installKeyboardActions and placing the Action in the SplitPane's ActionMap. Please refer to the key bindings specification for further details.

Creates a ActionListener for the JSplitPane UI that listens for specific key presses.

### createKeyboardEndListener

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
protected [ActionListener](http://docs.google.com/java/awt/event/ActionListener.html) **createKeyboardEndListener**()

**Deprecated.** *As of Java 2 platform v1.3.*

As of Java 2 platform v1.3 this method is no longer used. Subclassers previously using this method should instead create an Action wrapping the ActionListener, and register that Action by overriding installKeyboardActions and placing the Action in the SplitPane's ActionMap. Please refer to the key bindings specification for further details.

Creates a ActionListener for the JSplitPane UI that listens for specific key presses.

### createKeyboardResizeToggleListener

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
protected [ActionListener](http://docs.google.com/java/awt/event/ActionListener.html) **createKeyboardResizeToggleListener**()

**Deprecated.** *As of Java 2 platform v1.3.*

As of Java 2 platform v1.3 this method is no longer used. Subclassers previously using this method should instead create an Action wrapping the ActionListener, and register that Action by overriding installKeyboardActions and placing the Action in the SplitPane's ActionMap. Please refer to the key bindings specification for further details.

Creates a ActionListener for the JSplitPane UI that listens for specific key presses.

### getOrientation

public int **getOrientation**()

Returns the orientation for the JSplitPane.

### setOrientation

public void **setOrientation**(int orientation)

Set the orientation for the JSplitPane.

### isContinuousLayout

public boolean **isContinuousLayout**()

Determines wether the JSplitPane is set to use a continuous layout.

### setContinuousLayout

public void **setContinuousLayout**(boolean b)

Turn continuous layout on/off.

### getLastDragLocation

public int **getLastDragLocation**()

Returns the last drag location of the JSplitPane.

### setLastDragLocation

public void **setLastDragLocation**(int l)

Set the last drag location of the JSplitPane.

### getDivider

public [BasicSplitPaneDivider](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneDivider.html) **getDivider**()

Returns the divider between the top Components.

### createDefaultNonContinuousLayoutDivider

protected [Component](http://docs.google.com/java/awt/Component.html) **createDefaultNonContinuousLayoutDivider**()

Returns the default non continuous layout divider, which is an instanceof Canvas that fills the background in dark gray.

### setNonContinuousLayoutDivider

protected void **setNonContinuousLayoutDivider**([Component](http://docs.google.com/java/awt/Component.html) newDivider)

Sets the divider to use when the splitPane is configured to not continuously layout. This divider will only be used during a dragging session. It is recommended that the passed in component be a heavy weight.

### setNonContinuousLayoutDivider

protected void **setNonContinuousLayoutDivider**([Component](http://docs.google.com/java/awt/Component.html) newDivider,  
 boolean rememberSizes)

Sets the divider to use.

### getNonContinuousLayoutDivider

public [Component](http://docs.google.com/java/awt/Component.html) **getNonContinuousLayoutDivider**()

Returns the divider to use when the splitPane is configured to not continuously layout. This divider will only be used during a dragging session.

### getSplitPane

public [JSplitPane](http://docs.google.com/javax/swing/JSplitPane.html) **getSplitPane**()

Returns the splitpane this instance is currently contained in.

### createDefaultDivider

public [BasicSplitPaneDivider](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneDivider.html) **createDefaultDivider**()

Creates the default divider.

### resetToPreferredSizes

public void **resetToPreferredSizes**([JSplitPane](http://docs.google.com/javax/swing/JSplitPane.html) jc)

Messaged to reset the preferred sizes.

**Specified by:**[resetToPreferredSizes](http://docs.google.com/javax/swing/plaf/SplitPaneUI.html#resetToPreferredSizes(javax.swing.JSplitPane)) in class [SplitPaneUI](http://docs.google.com/javax/swing/plaf/SplitPaneUI.html)

### setDividerLocation

public void **setDividerLocation**([JSplitPane](http://docs.google.com/javax/swing/JSplitPane.html) jc,  
 int location)

Sets the location of the divider to location.

**Specified by:**[setDividerLocation](http://docs.google.com/javax/swing/plaf/SplitPaneUI.html#setDividerLocation(javax.swing.JSplitPane,%20int)) in class [SplitPaneUI](http://docs.google.com/javax/swing/plaf/SplitPaneUI.html)

### getDividerLocation

public int **getDividerLocation**([JSplitPane](http://docs.google.com/javax/swing/JSplitPane.html) jc)

Returns the location of the divider, which may differ from what the splitpane thinks the location of the divider is.

**Specified by:**[getDividerLocation](http://docs.google.com/javax/swing/plaf/SplitPaneUI.html#getDividerLocation(javax.swing.JSplitPane)) in class [SplitPaneUI](http://docs.google.com/javax/swing/plaf/SplitPaneUI.html)

### getMinimumDividerLocation

public int **getMinimumDividerLocation**([JSplitPane](http://docs.google.com/javax/swing/JSplitPane.html) jc)

Gets the minimum location of the divider.

**Specified by:**[getMinimumDividerLocation](http://docs.google.com/javax/swing/plaf/SplitPaneUI.html#getMinimumDividerLocation(javax.swing.JSplitPane)) in class [SplitPaneUI](http://docs.google.com/javax/swing/plaf/SplitPaneUI.html)

### getMaximumDividerLocation

public int **getMaximumDividerLocation**([JSplitPane](http://docs.google.com/javax/swing/JSplitPane.html) jc)

Gets the maximum location of the divider.

**Specified by:**[getMaximumDividerLocation](http://docs.google.com/javax/swing/plaf/SplitPaneUI.html#getMaximumDividerLocation(javax.swing.JSplitPane)) in class [SplitPaneUI](http://docs.google.com/javax/swing/plaf/SplitPaneUI.html)

### finishedPaintingChildren

public void **finishedPaintingChildren**([JSplitPane](http://docs.google.com/javax/swing/JSplitPane.html) jc,  
 [Graphics](http://docs.google.com/java/awt/Graphics.html) g)

Messaged after the JSplitPane the receiver is providing the look and feel for paints its children.

**Specified by:**[finishedPaintingChildren](http://docs.google.com/javax/swing/plaf/SplitPaneUI.html#finishedPaintingChildren(javax.swing.JSplitPane,%20java.awt.Graphics)) in class [SplitPaneUI](http://docs.google.com/javax/swing/plaf/SplitPaneUI.html)

### paint

public void **paint**([Graphics](http://docs.google.com/java/awt/Graphics.html) g,  
 [JComponent](http://docs.google.com/javax/swing/JComponent.html) jc)

Messaged to paint the look and feel.

**Overrides:**[paint](http://docs.google.com/javax/swing/plaf/ComponentUI.html#paint(java.awt.Graphics,%20javax.swing.JComponent)) in class [ComponentUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html) **Parameters:**g - the Graphics context in which to paintjc - the component being painted; this argument is often ignored, but might be used if the UI object is stateless and shared by multiple components**See Also:**[ComponentUI.update(java.awt.Graphics, javax.swing.JComponent)](http://docs.google.com/javax/swing/plaf/ComponentUI.html#update(java.awt.Graphics,%20javax.swing.JComponent))

### getPreferredSize

public [Dimension](http://docs.google.com/java/awt/Dimension.html) **getPreferredSize**([JComponent](http://docs.google.com/javax/swing/JComponent.html) jc)

Returns the preferred size for the passed in component, This is passed off to the current layoutmanager.

**Overrides:**[getPreferredSize](http://docs.google.com/javax/swing/plaf/ComponentUI.html#getPreferredSize(javax.swing.JComponent)) in class [ComponentUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html) **Parameters:**jc - the component whose preferred size is being queried; this argument is often ignored, but might be used if the UI object is stateless and shared by multiple components**See Also:**[JComponent.getPreferredSize()](http://docs.google.com/javax/swing/JComponent.html#getPreferredSize()), [LayoutManager.preferredLayoutSize(java.awt.Container)](http://docs.google.com/java/awt/LayoutManager.html#preferredLayoutSize(java.awt.Container))

### getMinimumSize

public [Dimension](http://docs.google.com/java/awt/Dimension.html) **getMinimumSize**([JComponent](http://docs.google.com/javax/swing/JComponent.html) jc)

Returns the minimum size for the passed in component, This is passed off to the current layoutmanager.

**Overrides:**[getMinimumSize](http://docs.google.com/javax/swing/plaf/ComponentUI.html#getMinimumSize(javax.swing.JComponent)) in class [ComponentUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html) **Parameters:**jc - the component whose minimum size is being queried; this argument is often ignored, but might be used if the UI object is stateless and shared by multiple components **Returns:**a Dimension object or null**See Also:**[JComponent.getMinimumSize()](http://docs.google.com/javax/swing/JComponent.html#getMinimumSize()), [LayoutManager.minimumLayoutSize(java.awt.Container)](http://docs.google.com/java/awt/LayoutManager.html#minimumLayoutSize(java.awt.Container)), [ComponentUI.getPreferredSize(javax.swing.JComponent)](http://docs.google.com/javax/swing/plaf/ComponentUI.html#getPreferredSize(javax.swing.JComponent))

### getMaximumSize

public [Dimension](http://docs.google.com/java/awt/Dimension.html) **getMaximumSize**([JComponent](http://docs.google.com/javax/swing/JComponent.html) jc)

Returns the maximum size for the passed in component, This is passed off to the current layoutmanager.

**Overrides:**[getMaximumSize](http://docs.google.com/javax/swing/plaf/ComponentUI.html#getMaximumSize(javax.swing.JComponent)) in class [ComponentUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html) **Parameters:**jc - the component whose maximum size is being queried; this argument is often ignored, but might be used if the UI object is stateless and shared by multiple components **Returns:**a Dimension object or null**See Also:**[JComponent.getMaximumSize()](http://docs.google.com/javax/swing/JComponent.html#getMaximumSize()), [LayoutManager2.maximumLayoutSize(java.awt.Container)](http://docs.google.com/java/awt/LayoutManager2.html#maximumLayoutSize(java.awt.Container))

### getInsets

public [Insets](http://docs.google.com/java/awt/Insets.html) **getInsets**([JComponent](http://docs.google.com/javax/swing/JComponent.html) jc)

Returns the insets. The insets are returned from the border insets of the current border.

### resetLayoutManager

protected void **resetLayoutManager**()

Resets the layout manager based on orientation and messages it with invalidateLayout to pull in appropriate Components.

### startDragging

protected void **startDragging**()

Should be messaged before the dragging session starts, resets lastDragLocation and dividerSize.

### dragDividerTo

protected void **dragDividerTo**(int location)

Messaged during a dragging session to move the divider to the passed in location. If continuousLayout is true the location is reset and the splitPane validated.

### finishDraggingTo

protected void **finishDraggingTo**(int location)

Messaged to finish the dragging session. If not continuous display the dividers location will be reset.

### getDividerBorderSize

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
protected int **getDividerBorderSize**()

**Deprecated.** *As of Java 2 platform v1.3, instead set the border on the divider.*

As of Java 2 platform v1.3 this method is no longer used. Instead you should set the border on the divider.

Returns the width of one side of the divider border.

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/BasicSplitPaneUI.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneDivider.VerticalDragController.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneUI.BasicHorizontalLayoutManager.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/plaf/basic/BasicSplitPaneUI.html)    [**NO FRAMES**](http://docs.google.com/BasicSplitPaneUI.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#3znysh7) | [FIELD](#2et92p0) | [CONSTR](#tyjcwt) | [METHOD](#3dy6vkm) | DETAIL: [FIELD](#2s8eyo1) | [CONSTR](#23ckvvd) | [METHOD](#32hioqz) |

[Submit a bug or feature](http://bugs.sun.com/services/bugreport/index.jsp)

For further API reference and developer documentation, see [Java SE Developer Documentation](http://docs.google.com/webnotes/devdocs-vs-specs.html). That documentation contains more detailed, developer-targeted descriptions, with conceptual overviews, definitions of terms, workarounds, and working code examples.

Copyright 2006 Sun Microsystems, Inc. All rights reserved. Use is subject to [license terms](http://docs.google.com/legal/license.html). Also see the [documentation redistribution policy](http://java.sun.com/docs/redist.html).